**A Review on Hybrid Analysis with Machine Learning in Android Malware Detection**

**Abstract:**

**Keywords:** Android Malware Detection, Hybrid Analysis, Machine Learning

**1. Introduction:**

//Stats android and malware

With the enormous growth of the android system [1], android malware also has grown significantly as well as upgraded its nature and activities [2]. Currently malware is smart enough to evade the typical anti-viruses or the obsolete approaches of malware detection. As android malware generally tries to preserve the facade of a benign application using multifarious evasion techniques, it is worthy and necessary to take a perceptive approach to defend them.

To detect android malware, there are three malware detection approaches: Static Analysis, Dynamic Analysis and Hybrid Analysis. Static analysis approach uses the static features of the android application such as Permissions, API Calls, Intents, Call-Graph, Opcode, Hardware Usage Analysis, Meta-data etc. Dynamic analysis approach investigates the dynamic behavior of the application running on an emulated environment or on a real device. These dynamic features/behaviors include System Call, Network Traffic, File Operations, Running Services, Network Operations etc. Hybrid analysis tries to incorporate both the static and the dynamic approach into a common ground.

Static and dynamic analysis have their own limitations. Currently malware authors are too smart to evade these detection techniques. They use many evasion techniques to evade the analysis. For static analysis the following evasion techniques are used by the malware authors: data obfuscation, control flow obfuscation, encryption, reflection, dynamically loaded code, repackaging etc. \cite {Tam}. For dynamic analysis, anti-analysis, mimicry, data obfuscation, misleading information flows and function indirections etc. are used as evasion techniques \cite {Tam}. Besides, limited code coverage lessens the effectiveness of the dynamic analysis.

As static and dynamic analysis have their weakness individually, combining both analysis into a common ground might be helpful in this regard. Hybrid analysis approach integrates both static and dynamic analysis to take advantages of their strengths and to mitigate their weakness. That’s why hybrid analysis so important to concentrate. Though hybrid analysis is complex enough, it is effective and feasible according to related researches. Previously, researchers have been more focused on static and dynamic analysis. As a result, there are a lot of research works are carried out in those domains, but comparatively a few works have been performed in hybrid analysis. Researchers nowadays give more focuses on it because of its effectiveness and potential.

Due to hybrid analysis approach’s huge potential and importance in android malware detection, there is a need for a brief review of the existing research works on hybrid analysis. In this work, we have provided a comprehensive review of hybrid analysis approach in android malware detection, analyzed the existing works: their strength and weakness and discussed about challenges, limitations and future directions in this regard.

To be specific, this work provides the following contributions:

1. To the best of our knowledge, this is the first review on hybrid analysis approach in android malware detection.
2. This work presents the importance of hybrid analysis over static and dynamic analysis by analyzing their weakness.
3. This work provides a thorough review of the existing works on hybrid analysis approach and an analysis of their pros and cons.
4. This work provokes a discussion about the challenges, limitations and future directions regarding hybrid analysis in android malware detection.

**2. Related Reviews:**

// this is the first…

// existing reviews….

**3. Background:**

**3.1 Android Malware:**

//What is malware….

Android malware is an application running on Android operating system that implicitly or explicitly performs malicious activities.

Android malware causes many malevolent things such as - disrupting normal functioning, taking access controls, leaking information, getting control of device without user’s knowledge etc. \cite {attia}.

//more stats…. Why it is alarming

Moreover, malware is growing exceedingly to keep pace with the immense growth of android applications. In each month, on average almost 10 million new malwares are introduced [18]. Most alarming thing is that, nowadays malware authors also aware of the malware detection system and they use many novel and crafty evasion techniques to avoid them. To fight against these cunning black hats, incorporating the most up-to-date and comprehensive detection technique is compulsory.

**3.2 Detection Techniques:**

Researchers analyze android malware generally with the following three approaches: Static, Dynamic and Hybrid Analysis. In static analysis, various static features are extracted from source code and meta-data. If the source code is not available, reverse engineering is applied to reproduce the source code. According to the static features, a detection model is built using machine learning technique/s to classify android malware. The most used static features in static analysis are as following: Permissions, API Calls, Intents, Call-Graph, Opcode, Hardware Usage Analysis, Meta-data etc.

Dynamic analysis deals with the dynamic features/behaviors of an application. To track the dynamic behaviors of an application, the application is to be run/executed on an emulated environment or on a real device. According to the dynamic features, a detection model is built using machine learning technique/s to classify android malware. The most used dynamic features in dynamic analysis are as following: System Call, Network Traffic, File Operations, Running Services, Network Operations etc.

Hybrid Analysis tries is to combine both static and dynamic features for detecting android malware. As it deals with both static and dynamic features at the same time, it is more complex and costly with respect to time and effort. May be this is the reason for the fact that, there is not a good number of researches on hybrid analysis in comparison with the static or dynamic analysis.

**3.3 Limitations of Static and Dynamic Analysis:**

Static Analysis faces many troubles such as data obfuscation, control flow obfuscation, encryption, reflection, dynamically loaded code, repackaging etc. \cite {Tam} by the shrewd malware authors.

//any more description needed on those limitations?

On the other hand, Dynamic Analysis also have some drawbacks. To evade dynamic analysis, anti-analysis technique is used frequently by the malware authors to detect virtual machines or emulated environments. If the application detects emulated environments in advance, they will act like benign application. By doing so, dynamic analysis might be failed to detect android malware. Besides, malware authors use mimicry, data obfuscation, misleading information flows and function indirections etc. to evade dynamic analysis \cite {Tam}. The biggest weakness of dynamic analysis is limited code coverage: covering all path is not feasible when investigating dynamic behavior of an application.

**4. Hybrid Analysis Approach:**

Hybrid analysis integrates both static and dynamic features for more effectiveness than the static or dynamic analysis. Firstly, this hybrid approach seeks to extract the static features and dynamic features of android applications. After that, those extracted static and dynamic features are combined to build a detection model. Extracting and combining static and dynamic features is the most challenging task of the hybrid analysis. Finally, according to the combined static and dynamic features, a detection model is built using machine learning technique/s to classify android malware.

By incorporating static and dynamic approach into a common ground, hybrid analysis leads more complexity in android malware detection. The detection process is more likely to take more time and effort. Though hybrid approach might be more effective in android malware detection than the static or dynamic approach, accomplishing a viable malware detection technique is effortful.

As hybrid approach is the combination of static and dynamic approach, this approach can overcome the individual weakness as well as can accumulate the advantages of them. Thereby, hybrid approach strengthens the detection process with the cost of time and complexity. Hybrid methods can also increase robustness, monitor edited apps, increase code coverage and find vulnerabilities \cite {Tam}.

**5. Literature Review:**

Despite there is not so many researches are carried out in android malware detection using hybrid analysis, those few researches in this regard exhibit better performance on average than the static and dynamic approaches and engender a lot of opportunities. In the following section we present a inclusive review of the consequential researches in hybrid analysis.

//tools

One of the state-of-the-art work in this regard, Mobile SandBox [12] used 7 static features and investigated Native Code and Network Traffic as dynamic features to classify malware despite lacking solid performance metrics.

Marvin [13] employed a lot of static and dynamic features to detect android malware. It extracts though it ignores famous dynamic feature- ’System Call’ in its approach.

Samadroid [14] incorporated ’System Call’ and outperform Drebin, but it used old dataset.

Kapratwar et al. [15] used ’Permissions’ and ’System Calls’ for Hybrid Analysis, even though its performance was poor.

Talha et al. [16] revealed unknown characteristics of Android malware, however it did not integrate any machine learning approach.

Hadm [17] subsumed Deep Neural Network for feature extraction from a set of static and dynamic features.

Dhanya \cite {37} used Permissions as static and API Calls as dynamic feature. Separability assessment Criteria is used for feature selection in this research. Using the 77 selected features and four different machine learning algorithms (Naïve Bayes, SVM, J48 & Random Forest) they evaluated their work. Their performance regarding F-measure, precision and recall is dubitable (given in Table 1) as they used Drebin, an outdated and limited dataset. Besides they did not consider any other features except Permissions and API Calls.

Liu \cite {28} proposed a hybrid malware detecting scheme for android where Permissions and APIs are used as static features and System Calls used as dynamic features. Their scheme’s detection accuracy is to 93.33%∼99.28% according to experimental results. Though they considered only a small feature-set and their dataset is also limited.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Ref.** | **Publish ing Year** | **Static Features** | **Dynamic**  **Features** | **Dataset Source** | **Dataset Size** | **Algori thms** | **Metrics** | **Values** | **Limitation** |
| Dhanya \cite {37} | 2019 | Permissions | API Calls | Drebin | 400 apps (200: 200) | Naïve Bayes, SVM, J48 & Random Forest | F-measure, Precision, Recall | 0.71 ~0.975, 74.7%~97.6%, 72.5%~97.5% | Limited and Old Dataset, Few Features |
| Liu \cite {28} | 2016 | Permissions, API | System Calls | Gnome Project, Wandoujia App Market | 1000 apps (1000: 1000) | SVM, KNN | ACC,  TPR,  FPR | 93.33%~99.28%,  94.59%~99.47%,  0.20%~11.01% | Limited Dataset, Few Features |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

//Hydroid,

//Hybridroid

**6. Discussion:**

// more research is needed in this regard:

As mentioned before, there is not enough researches in hybrid analysis, though it is a promising and effective approach in android malware detection. So, researchers have to put more emphasize in this regard. A lot of opportunities and research directions are available right now. Researchers enthusiastic focus in this field might have been beneficial from android application security perspective as well as from academia perspective. To fight against the ill-minded and escalating malware authors community, more researches is essential.

//dataset inadequacy:

Though malware is growing enormously in every second, there does not exist any up-to-date dataset for the researchers. Previously stated, almost 10 million new malwares are found in each month [18]. But most of the dataset used in research is dated and obsolete nowadays. Thereby, their performance in android malware detection is doubtful considering the vast population of the new malware. Dataset inadequacy is a vital factor as dataset is responsible for the evaluation of any research. So, android malware dataset has to be updated on regular basis to assure the effectiveness of the new researches and to justify the feasibility of the existing researches.

//new malware family

// any new feature?

// hybrid complexity

Since hybrid approach combines static and dynamic approach, its overall complexity is higher with respect to time, cost and effort. How to reduce the complexity of hybrid analysis can be a potential direction for future researchers. The outcome can be a more viable and effective approach.

**7. Conclusion:**

Android Malware is the key factor for the most security breaches in android operating system. Detecting android malware effectively and feasibly in advance is the biggest challenge of this fast-growing digital world. Hybrid analysis approach has the capability and can offer a sound direction in this regard. By exploring this field, researchers have published some remarkable researches already. This work tends to highlight those researches by providing a comprehensive review of those works. Besides it points out the specific challenges, limitations and future directions in hybrid analysis. By doing so, this research seeks to contribute to the academia as well as raise concern for android mobile application security.

**References:**